

METHOD FOR SYNCHRONIZING DATA UTILIZED IN REDUNDANT,
CLOSED LOOP CONTROL SYSTEMS

BACKGROUND

[0001] The present disclosure relates generally to vehicle chassis control systems and, more particularly, to a method for synchronizing data utilized by redundant, closed-loop feedback control systems implemented in distributed vehicle chassis systems.

[0002] Chassis control systems such as, for example, electronically controlled brakes and steering, typically rely on feedback control methods to provide desired system performance and stability. Such systems utilize input sensors and feedback sensors to accomplish the feedback control, and are generally locally implemented within an individual electronic control unit (ECU). In this configuration, the sensors and associated computer signals are locally available at the ECU to perform the closed loop control. Figure 1 is a block diagram illustrating an example of the general structure of a closed loop control system, which further includes a feedforward input coupled to the basic feedback structure. The diagram in Figure 1 represents the most general, high-level description of a feedback/feedforward control system commonly implemented in automotive control systems.

[0003] Traditionally, these control systems have been included in vehicle architecture designs that are centralized in nature (i.e., designs that may be characterized as modular or self-contained). However, a growing motivation in the industry is to provide a distributed control architecture in which various control functions are distributed across multiple ECU's. In so doing, the level of vehicle subsystem integration is increased in an effort to reduce overall electrical system cost, improve packaging, reduce mass, increase availability, etc.

[0004] With certain vehicle systems, such as steer-by-wire and brake-by-wire, redundant control systems may be utilized to achieve a desired level of

10043930.01102

system fault tolerance. For example, multiple actuators, sensors and ECU's may be incorporated into the system to allow for continued operation of the brakes and/or steering in the event of a failure of some of the system components. Certain difficulties exist, however, in attempting to operate a redundant, closed loop feedback system in both a parallel and synchronous manner within the framework of a distributed control architecture.

SUMMARY

[0005] The above discussed and other drawbacks and deficiencies of the prior art are overcome or alleviated by a method for synchronizing data utilized in a redundant, closed-loop feedback control system. In an exemplary embodiment, the method includes configuring a plurality of control nodes within the control system, with each of the plurality of control nodes transmitting and receiving data through a common communication bus. At each of the plurality of control nodes during a given control loop time $T = N$, the receipt of externally generated data with respect to each control node is verified, the externally generated data having been generated during a preceding control loop time $T = N-1$. At each of the plurality of control nodes during the given control loop time $T = N$, output control data is calculated using the externally generated data. During the given control loop time $T = N$, the calculated output control data from each individual control node is further transmitted over the communication bus to be later utilized by other control nodes during a subsequent control loop time $T = N+1$.

[0006] In a preferred embodiment, at each of the plurality of control nodes during the given control loop time $T = N$, reference input data is calculated using the externally generated data received during the preceding control loop time $T = N-1$. In addition, at each of the plurality of control nodes during the given control loop time $T = N$, local sensor inputs are acquired and transmitted through the communication bus to be used by other control nodes during the subsequent control loop time $T = N+1$. The local sensor inputs acquired at each of the plurality of control nodes during the given control loop time $T = N$ are further used in calculating output control data during the

subsequent control loop time $T = N+1$.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] Referring to the exemplary drawings wherein like elements are numbered alike in the several Figures:

[0008] Figure 1 is a block diagram illustrating an example of the general structure of a closed loop control system, which further includes a feedforward input coupled to the basic feedback structure;

[0009] Figure 2 is a schematic diagram illustrating an example of a redundant electromechanical feedback control system configured within a distributed electrical architecture;

[0010] Figure 3 is a block diagram of the redundant feedback control system of Figure 2, shown in operation with a primary set of input and feedforward sensors;

[0011] Figure 4 is a block diagram of the redundant feedback control system of Figure 2, shown in operation with a secondary (backup) set of input and feedforward sensors;

[0012] Figure 5 is a block diagram of the redundant feedback control system of Figure 2, shown operating with a hybrid (both primary and backup) combination of input and feedforward sensors; and

[0013] Figure 6 is an algorithm flow diagram illustrating a method of a synchronizing system data to be utilized across a network of electronic control units, in accordance with and embodiment of the invention.

DETAILED DESCRIPTION

[0014] Referring initially to Figure 2, there is shown a schematic diagram illustrating an example of a redundant electromechanical feedback system 10 configured within a distributed electrical architecture. In such a parallel-operated redundant system 10, both control actuators 12 (i.e., motor 1 and motor 2) are intended to be activated and operated simultaneously. The combination of both actuator outputs therefore provides a net control action.

[0015] The exemplary system architecture includes certain desired node

characteristics, wherein the “nodes” are embodied as electronic control units (ECU’s) 14 that perform a certain function or functions, and are interconnected through a data communication bus 16. For example, there is shown a redundant sensor distribution embodied throughout nodes 1-4, as well as a redundant actuator command distribution embodied within motor 1 and motor 2. Specifically, control nodes 3 and 4 are intended to provide simultaneous actuator (i.e., motor) control (both magnitude and time synchronized), while all of the nodes (sensing nodes 1 and 2, and control nodes 3 and 4) are further intended to provide fail silent behavior. Thus, each node (ECU 14) should both perform its intended function and provide data to the communication bus 16. If it determined that an ECU 14 cannot perform its intended function(s), the output thereof should be disabled and/or the conditioned sensor signals provided to the communication bus 16 should be ceased.

[0016] Whenever dual feedback systems are operated in parallel, any performance variability between the two subsystems should preferably be minimized, since the performance variability from one subsystem could be interpreted as a disturbance by its counterpart subsystem. Otherwise, undesirable overall system performance may result. One source of performance variability between the individual subsystems can be system latency and noise, contributed by the use of control signal and sensors used by the feedback control.

[0017] A chassis control system architected to operate as a distributed system will utilize a data communications bus to facilitate the exchange of system data, commands, and status or health among the various ECU nodes. In order to provide the desired minimization of performance variations, while also maintaining system availability in the presence of faults, the system should synchronize the use of a set of common calculated commands, sensors and status data. The calculated and sensed data should then be synchronized in both magnitude and time. Thus, a distributed system for an automotive vehicle that implements redundancy principles to satisfy overall system fault tolerance faces the challenge of composing the sensor and actuators in such a way as to satisfy system power efficiency, immunity to electrical noise, and physical packaging.

Various examples of possible redundant control system operating configurations are presented in Figures 3-5. Figure 3 is a block diagram of the dual, redundant control system 10 of Figure 2, wherein the actual control is implemented using primary set of input and feedforward sensors (i.e., Ref Input 1, Feedforward Input 1), as well as a primary feedback sensor (Feedback Sensor 1). The signals on Ref Input 1 and Feedback Sensor 1 are internal to control node 3, but are also provided to control node 4 through communication bus 16. However, the signal on Feedforward Input 1 is provided to both control nodes 3 & 4 by sensing node 1 through communication bus 16.

[0018] In Figure 4, the redundant control system 10 is shown operating with a secondary set of input and feedforward sensors (i.e., Ref Input 2, Feedforward Sensor 2), as well as a secondary feedback sensor (Feedback Sensor 2). In this configuration, the signals on Ref Input 2 and Feedback Sensor 2 are internal to control node 4 and are further provided to control node 3 via communication bus 16. The signal on Feedforward Input 2 is provided to both control nodes 3 & 4, by sensing node 2, via communication bus 16.

[0019] Finally, Figure 5 illustrates the redundant control system 10 operating in a hybrid configuration, utilizing a combination of primary and secondary sensor inputs, due to a condition such as (for example) the detection of a failed primary feedback sensor (Feedback Sensor 1). Thus configured, the system 10 uses the primary reference and feedforward inputs, while also using the secondary or backup feedback input.

[0020] As described earlier and illustrated in Figure 2, a given control system may dictate that certain nodes be dedicated for sensor acquisition, while others are dedicated to actuator or motor control. Accordingly, a method is needed such that the sensor data and system status from the sensing nodes are made available to the actuator nodes, while also providing minimal system latency. Additionally, such a method should also provide for the condition that each parallel redundant actuator node performing feedback control utilize common data.

[0021] Therefore, in accordance with an embodiment of the invention, there is disclosed a method for synchronizing system data across a network of

ECU nodes, such that each feedback subsystem will have access to and use the same reference control inputs and feedback sensors, even in the presence of a detected system fault(s). The method 100 for synchronizing system data across the network of ECU nodes is illustrated, in one embodiment, by the algorithm flow diagram shown in Figure 6. It will be noted that the steps outlined therein are preferably implemented at each ECU node in the system. Furthermore, method 100 is implemented within the framework of a control loop beginning at a time $T = N$, and progressing through sample times $N+1$, $N+2$, ...etc., as shown in block 101.

[0022] Proceeding to block 102, method 100 verifies the receipt of external calculation data at each control node, generated from other participating system nodes (via communication bus 16). All external data directly accessible by a particular control node will have been acquired and stored in a storage medium 104 (e.g., a random access memory) at time $T = N-1$ (i.e., the previous control sample time). For example, in order to ultimately compute output 1, control node 3 will use the data from one of the two external feedforward sensors (Feedforward Sensor 1 or Feedforward Sensor 2), depending upon the particular system configuration. Because this feedforward data is transferred and received over the communication bus 16 (as indicated by block 106) during the previous control loop, the data is already stored in storage medium 104 and ready to be verified during the next control loop. Therefore, at block 102, each node verifies the receipt of all stored data that originated externally with respect to that particular node.

[0023] Proceeding to block 108, method 100 then directs the control nodes to calculate updated input reference commands (Ref Input 1 and Ref Input 2) to be used in calculating the feedback control output commands during the next control sample $T = N+1$. As can be seen, the updated Ref Input 1 and Ref Input 2 are calculated using data stored during $T = N-1$. Once calculated, the updated input reference commands are both stored in memory 104 and transmitted to other nodes through data communication bus 16, as shown at block 110.

[0024] Next, at block 112, the output actuator command for time $T = N$

is calculated at each control node, using externally received data, stored sensor data and input reference commands from time $T = N-1$. It will be noted that in calculating the output actuator commands, one of the input reference commands will be used. However, as stated previously, the input reference command so used is the not the one just calculated in block 108 for $T = N$, but the one previously calculated in block 108 for $T = N-1$. As is the case with the newly calculated reference input, the new command actuator outputs are also stored in memory and transmitted over communication bus 16. Method 100 thereafter proceeds to block 114, where data from sensors locally connected to each node is acquired and conditioned for time $T = N$. This locally acquired sensor data is then stored for use by other nodes in the next control loop $T = N+1$.

[0025] It will further be noted that method 100 also provides for execution of error detection algorithms concurrently with each of the above described steps for a given control loop, as schematically depicted by block 116 and the shaded blocks behind blocks 102, 108, 112 and 114. The error detection algorithms 116 are preferably designed to isolate any faults specific to a particular functional task to be performed. More specifically, algorithms 116 use externally received data from other ECU nodes (the data also including system health status), as well as local sensor and input command calculations to determine the configuration of control inputs, feedback sensors, and output actuator commands to be used by each node in future control loops. Thus, locally acquired sensing inputs, calculated commands, diagnosis of the local variables, and the current local ECU “health” are all provided to the communication bus 16 for transmission to the other ECU’s in the system 10.

[0026] The selection of sensor configurations for the closed loop control under the presence of system errors is determined based on an “a priori” design of the decision tree (or logic) pre-programmed at each node. Switching from one signal configuration to an alternate desired configuration is performed once global confirmation is received by the system nodes. The proper operation of the error detection algorithms at each node will allow the redundant system 10 to dynamically configure its control to allow for a variety of input and output configurations, as the examples of Figures 3-5 illustrate.

[0027] Moreover, the error detection algorithms 116 provide that all system nodes are coordinated at each successive control loop to have the desired configuration of inputs, feedback sensors, and output commands. By way of example, some specific tasks that may be executed by the error detection algorithms 116 include, but are not limited to: performing local sensor diagnosis (e.g., determining a valid range, rate of change, etc.); performing sensor comparison checks with other local and externally received signals (via communication bus 16); determining future sensor and command signal control loop configurations using voting logic, or other known diagnostic methods; updating local system health status information; and determining a desired actuator output condition (e.g., fail silent: yes/no).

[0028] Finally, at block 118, method 100 also allows for the execution of various background ECU tasks, such as parity checking or other related microprocessing functions, prior to the next control process interrupt. Thereafter, method 100 returns to block 101 to repeat the above described steps after a process interrupt occurs.

[0029] It is to be understood that the method 100 embodied and described herein is further capable of performing synchronization functions for ECU's utilized in systems of greater than double redundancy. That is to say, the principles of method 100 may be further expanded and scaled to systems having an even greater number of sensor and actuator nodes. Through the use of the above described method, several advantages are realized. First, the maximum system latency to transmit updated output commands to the actuator(s) is two control loop samples, which latency is further independent of the number of system control ECU's or input/feedback sensors utilized. Second, the extent of any performance variation between actuator subsystems is reduced to only those variations related to the actual physical plant dynamics of the actuator devices under closed loop control. Third, a generated control command is calculated based on the same input and feedback sensors. Thus, the servo control loops will benefit from improved coordinated performance. This is especially the case where the sensors are susceptible to external noise, as is common with many analog-conditioned sensors.

[0030] In addition, the disclosed invention may be embodied in the form of computer-implemented processes and apparatuses for practicing those processes. The present invention can also be embodied in the form of computer program code containing instructions embodied in tangible media, such as floppy diskettes, CD-ROMs, hard drives, or any other computer-readable storage medium, wherein, when the computer program code is loaded into and executed by a computer, the computer becomes an apparatus for practicing the invention. The present invention can also be embodied in the form of computer program code, for example, whether stored in a storage medium, loaded into and/or executed by a computer, or as data signal transmitted whether a modulated carrier wave or not, over some transmission medium, such as over electrical wiring or cabling, through fiber optics, or via electromagnetic radiation, wherein, when the computer program code is loaded into and executed by a computer, the computer becomes an apparatus for practicing the invention. When implemented on a general-purpose microprocessor, the computer program code segments configure the microprocessor to create specific logic circuits.

[0031] While the invention has been described with reference to a preferred embodiment, it will be understood by those skilled in the art that various changes may be made and equivalents may be substituted for elements thereof without departing from the scope of the invention. In addition, many modifications may be made to adapt a particular situation or material to the teachings of the invention without departing from the essential scope thereof. Therefore, it is intended that the invention not be limited to the particular embodiment disclosed as the best mode contemplated for carrying out this invention, but that the invention will include all embodiments falling within the scope of the appended claims.